Amendments to the Claims:

A listing of the entire set of pending claims (including amendments) is submitted herewith per 37 CFR 1.121. This listing of claims will replace all prior versions, and listings, of claims in the application.

1. (Currently amended) A method for encrypting a digital data stream in a transmission system that uses orthogonal codes for the modulation, the method comprising:

constructing a k^{th} connection for a k^{th} digital data stream ($d^{(k)}$) by a k^{th} transmitter,

mixing the digital data stream ($d^{(k)}$) of the transmitter with a spreading code that is assigned to this k^{th} connection,

assigning different spreading codes ($g_1^{(k)}$, $g_2^{(k)}$ ... $g_H^{(k)}$) from a defined set (G_i), wherein the spreading codes are produced decentrally, and

producing a transmission signal ($s^{(k)}$) through the mixing,

wherein the degree of encryption of the k^{th} digital data stream ($d^{(k)}$) is increased during the k^{th} connection through an allocation of a sequence for the application of the different spreading codes ($g_1^{(k)}$, $g_2^{(k)}$ ... $g_H^{(k)}$) and/or a hop interval (I_{hop}) by the k^{th} transmitter.

2. (Previously presented) The method as claimed in claim 1, further comprising defining the sequence of the application of the content of a set of spreading codes (G_i) with a permutation function (S_i) by stating the position ({p_1, p_2 ... p_M}).

3. (Currently amended) A method for encrypting a digital data stream that is to be transmitted in a transmission system, wherein after a connection set-up, necessary parameters for transmission and recovery are transmitted, the method comprising:

communicating an encryption key and thus:

establishing a permutation function (S_i) that defines a sequence of the application of the content of a set of spreading codes, wherein the spreading codes are produced decentrally,

establishing a set (G_i) of spreading codes, and/or

establishing a hop interval (I_{hop}),

wherein the establishing a permutation function (S_i), the establishing a set (G_i) of spreading codes, and/or establishing a hop interval (I_{hop}) can be carried out in any order.

4. (Currently amended) A method for encrypting a digital data stream in a transmission system, the method comprising executing a first permutation procedure which contains a loop with the following steps:

setting an interval (n) to "1";

waiting for the end of a predefined hop interval (I_{hop});

increasing the interval (n) by the value 1;

carrying out a comparison to see whether the current value of the interval (n) is greater than the total number (M) of the elements of a permutation function (S_i) which states the positions of the spreading code (g_n) of a set (G_i) of spreading codes that is to be used for encrypting the digital data stream, wherein the spreading codes are produced decentrally, wherein alternatively the following takes place:

if the comparison has a positive result:

resetting of the interval (n) to "1";

if the comparison has a negative result:

equating the current spreading code (g_n) with the spreading code (g_{p_n}) that stands at the position (p_n) stipulated by the permutation function (S_i).

5. (Previously presented) A device for carrying out the method as claimed in claim 1, wherein the device has a first code generator that creates the respectively current spreading code (g_n).

6. (Currently amended) A method for decoding a received digital data stream that was sent encrypted in a transmission system, the method comprising executing a second permutation procedure that contains a loop including:

setting an interval (n) to "1";

waiting for the end of a predefined hop interval (I_{hop});

increasing the interval (n) by the value 1;

carrying out a comparison to see whether the current value of the interval (n) is greater than the total number (M) of the elements of a permutation function (S_i) which states the positions of the spreading code (g_n) of a set (G_i) of spreading codes that is to be used for decoding the encrypted digital data stream, wherein the spreading codes are produced decentrally, wherein alternatively the following takes place:

if the comparison has a positive result:

resetting of the interval (n) to "1";

if the comparison has a negative result:

equating the current spreading code (g_n) with the spreading code (g_{p_n}) that stands at the position (p_n) stipulated by the permutation function (S_i).

7. (Previously presented) A device for carrying out the method as claimed in claim 6, wherein the device has a second code generator that produces the current spreading code (g_n).

8. (Previously presented) A transmission system that uses orthogonal codes for the modulation, with a first device for encrypting a digital data stream as claimed in claim 5, wherein the digital data stream ($d^{(k)}$) is mixed with a spreading code, and with a second device for decoding a digital data stream that was sent encrypted, wherein the second device executes a second permutation procedure that contains a loop with the following steps:

setting an interval (n) to "1";

waiting for the end of a predefined hop interval (I_{hop});

increasing the interval (n) by the value 1;

carrying out a comparison to see whether the current value of the interval (n) is greater than the total number (M) of the elements of a permutation function (S_i) which states the positions of the spreading code (g_n) of a set (G_i) of spreading codes that is

to be used for decoding the encrypted digital data stream, wherein alternatively the following takes place:

if the comparison has a positive result:

resetting of the interval (n) to "1";

if the comparison has a negative result:

equating the current spreading code (g_n) with the spreading code (g_{p_n}) that stands at the position (p_n) stipulated by the permutation function (S_i),

wherein the system has means for

carrying out encryption,

carrying out decoding of a digital data stream that was transmitted encrypted.

9. (Cancelled)